

REMARKS

Claims 1-6, 8-16, and 18-24, all the claims pending in the application, stand rejected on prior art grounds. Claims 1, 11, and 21 are amended herein. Applicants respectfully traverse these rejections based on the following discussion.

I. The Prior Art Rejections

Claims 1-6, 8-16, and 18-24 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Dan, et al. (U.S. Publication No. 2002/0178103), hereinafter referred to as Dan, in view of Thomas (U.S. Publication No. 2003/0167446). Applicants respectfully traverse these rejections based on the following discussion.

Dan teaches a method for automating contract negotiation between a plurality of parties over a communications network. The parties communicate and agree upon a negotiation protocol before commencing the negotiation in a meta contract that is formed to govern or control the negotiation process. The automatic negotiation may include at least one sub negotiation. Machine-executable rules are specified to enable an automatic negotiation to take place between servers over a communications network. A successful negotiation may result in the formation of an electronic commerce contract. Each party may maintain the contract state of the overall negotiation, which may take place among two or more parties, wherein at least one party may be represented by a broker. Thus, complex negotiations may be handled automatically by the inventive method. The negotiation may be conducted semi-automatically to allow for human intervention in the negotiation process.

Thomas teaches a method of recording changes to a markup language file which employs

application-defined tags. The changes are recorded in a delta file which is also a markup language file providing validation of the recorded changes against substantially the same markup language structure as that of the markup language file being changed. Where the original markup language file is an XML file with a DTD, a DTD can be created for the delta file which substantially follows the DTD of the original markup language file. Strict compliance of the data recorded in the delta file with the delta DTD provides validation of the changes with respect to the original XML file.

However, the Applicants' claimed invention has features not taught or suggested by Dan and Thomas. In particular, amended independent claims 1, 11, and 21 provide, in part, "...pre-building static structures of said XML transaction, wherein said static structures comprise a pre-built XML data structure with pre-filled values based on a transaction type of said XML transaction and a predetermined trading partner profile; ... combining said static structures with said dynamic structures at a runtime of said XML transaction based on said sequence, said type of XML transaction, said trading partner profile, and said dynamic structures of said XML transaction, wherein an occurrence of said runtime of said XML transaction occurs when said XML transaction is sent to a trading partner, wherein said combining comprises filling the empty tags of said dynamic structures; and constructing a final XML structure based on the combining process, wherein said final XML structure comprises fully built dynamic structures that comprise completely filled tags, and wherein said final XML structure is validated by comparing said final XML structure against said copy of said original data type definition format for said XML transaction." These features are neither taught nor suggested in Dan and Thomas. In fact, as demonstrated below, it appears that the Office Action is misconstruing Dan's "almost-complete

electronic contract document with a few fields left blank.”

Even if Dan were properly combined with Thomas, they would still fail to teach the Applicants’ claimed invention because of the occurrence of the time of occurrence of the combining of the static and dynamic structures occurs at the runtime (i.e., when said XML transaction is sent to a trading partner) of the XML transaction, whereas in Dan this occurs during the contract negotiation time (prior to the runtime of the XML transaction). Pages 4 and 20 of the Office Action points to paragraphs 33 and 34 of Dan of teaching the Applicants’ “combining said static structures with said dynamic structures at a runtime of said XML transaction based on said sequence, said type of XML transaction, said trading partner profile, and said dynamic structures of said XML transaction, wherein an occurrence of said runtime of said XML transaction occurs when said XML transaction is sent to a trading partner.” However, a closer review of the cited sections of Dan reveals no such teaching.

Paragraphs 33 and 34 of Dan recite:

[0033] The TPA template or party profile may be included as part of the information advertised by the service provider in step 60 of FIG. 2. The profile serves as the starting point of a negotiation by providing an initial version of a contract document. The profile may include information such as: products and services provided, specific business processes that the service provider can perform, security requirements, and technology information such as which message-exchange protocols are supported by the service provider. The service provider’s profile may be embodied in a variety of different forms. Several examples of the service provider’s profile are described herein, although alternative profile forms will be apparent to those of ordinary skill in the art.

[0034] In one embodiment, the service provider’s profile may describe the capabilities of one party. This profile may be expressed, for example, as an XML document whose contents may be incorporated into a contract. The information contained in the profile may include not only the capabilities of a party but also

may contain requirements of the interacting party in the form of a contract template. The contract template is provided to express a contract either between a pair of roles or between an actual party (whose profile is represented by the template) and a role. One example of a contract template is an almost-complete electronic contract document with a few fields left blank: these fields are to be filled in by the negotiating party. An enhanced template additionally specifies, in an associated document, the acceptable choices for the negotiable fields.

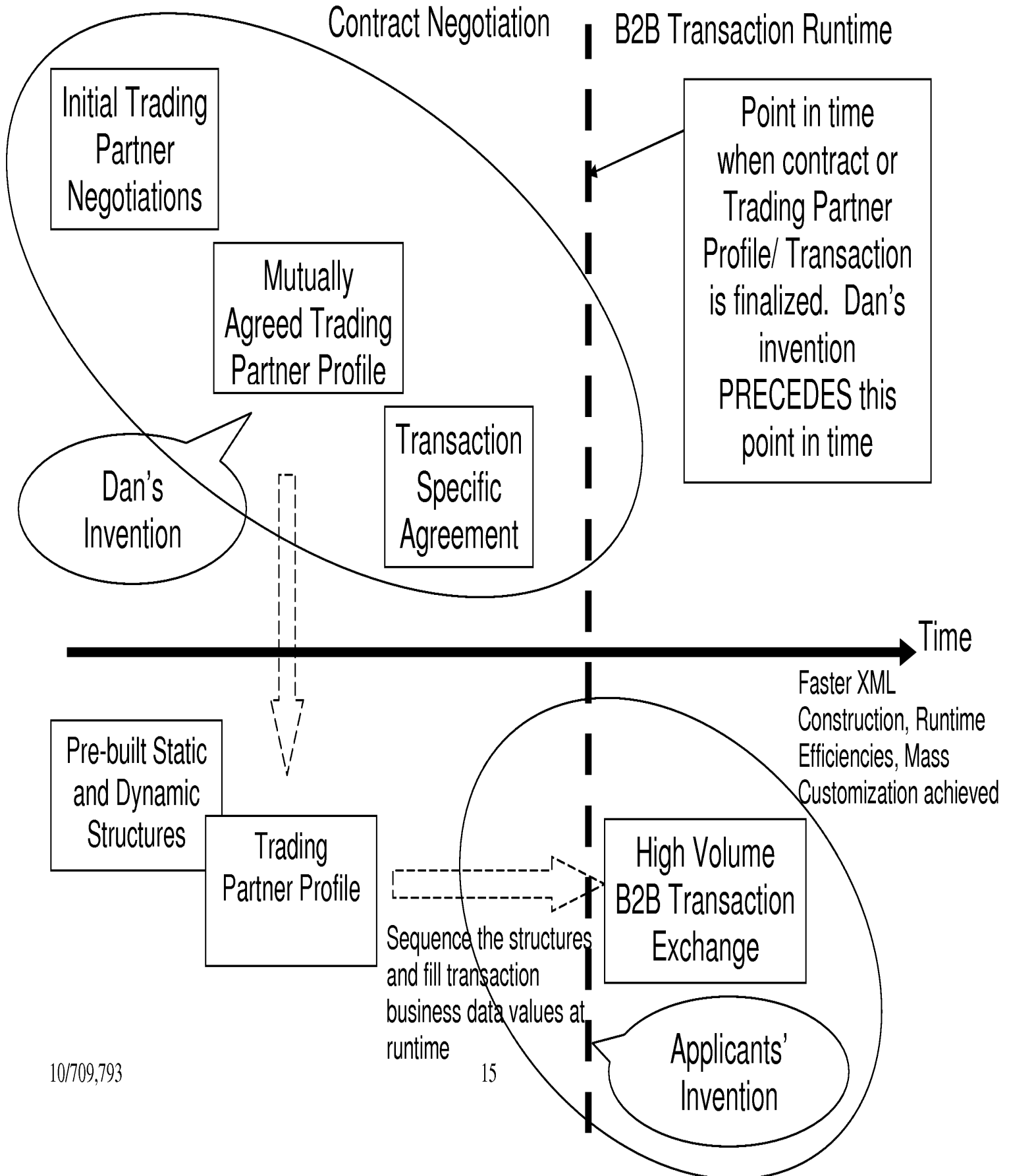
The above language refers to “[o]ne example of a contract template is an almost-complete electronic contract document with a few fields left blank.” Clearly, Dan is referring to one type of contract document (i.e., a document with a few fields left blank) and refers to this type of document as “an almost-complete electronic document.” In other words the phrase “an almost-complete electronic document” serves as an adjective describing the type of contract document of Dan with the phrase “with a few fields left blank” further describing what constitutes “an almost-complete electronic document.” Whereas, the Office Action in its conclusion, “[t]he examiner further wishes to state that the initial contract must combine the static fields (almost complete portions) and the dynamic fields (the blank portions) at runtime (when the contract is sent to other party)” erroneously argues that the contract contains two types of fields (an almost complete portion and a blank portion). As demonstrated above Dan teaches a contract that contains blank fields, whereas the Applicants’ claimed invention provides “fully built dynamic structures that comprise completely filled tags.” Accordingly, the Applicant’s claimed invention is patentable over Dan even if combined with Thomas.

Furthermore, in Thomas the constructed XML is compared to a pre-established DTD and if there is a difference (delta) between the constructed XML and the pre-established DTD, then Thomas changes the DTD (and saves these changes into the DTD) (see Figure 3 of Thomas).

Conversely, in the Applicants' claimed invention the constructed XML is compared to a copy of the pre-established DTD and if there is a difference between the constructed XML and the copy of the pre-established DTD, then the XML is invalidated (wherein said final XML structure is validated by comparing said final XML structure against said copy of said original data type definition format for said XML transaction). Therefore, in the Applicants' invention if a difference exists, then the DTD is not changed, but rather the process is repeated until no changes exist. Page 21 of the Office Action states that the Applicants rely on "if there is a difference between the constructed XML and the copy of the pre-established DTD, then the XML is invalidated" to distinguish its claims from Thomas, but the Office Action indicates that this language is not included in the rejected claims. However, the Applicants have indicated in parenthetical form which claimed language specifically teaches this concept (wherein said final XML structure is validated by comparing said final XML structure against said copy of said original data type definition format for said XML transaction). Therefore, this limitation is provided in the Applicants' claims.

In other words, to use a simple analogy, if the DTD can be likened to an answer key for an exam, and the constructed XML is a student's response to an exam, then in Thomas, if there are differences between the answer key and the student's response, then the answer key is changed. However, in the Applicants' invention, if there are differences between the answer key and the student's response, then the student's response is deemed invalidated (i.e., incorrect). The differences between the Applicants' invention and Dan can be graphically shown as follows:

Context – Typical timeline in Dan vs. Applicants' Invention



Additionally, Dan handles XML as a form template to be filled:

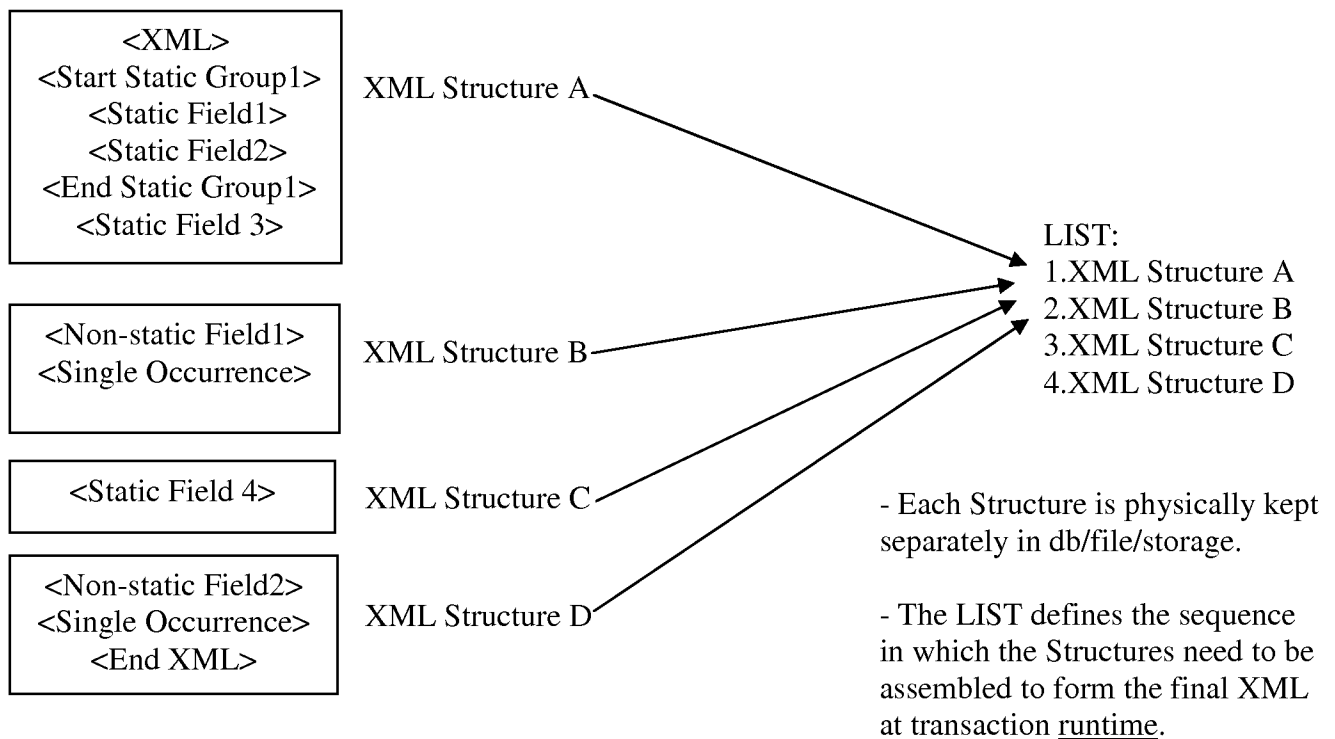
```
<XML>
<Start Static Group1>
  <Static Field1>
  <Static Field2>
<End Static Group1>
<Static Field 3>
<Non-static Field1>
<Static Field 4>
<Non-static Field2>
<End XML>
```

} Values are filled in for static sections of XML

← “Empty” tags are left for non-static/“negotiable” fields

Thus, Dan uses a form template with some values filled in and some values left blank.

Accordingly, there is no intelligence in Dan’s approach to differentiate if a field is single occurrence, multiple occurrence, etc. Conversely, the Applicants’ break the XML into pieces and string them together via an external list:



The Applicants provide for pre-building of static structures of an XML transaction. The Office Action suggests that Dan teaches this as follows:

- The profile serves as the starting point of a negotiation by providing an initial version of a contract document (paragraph 30).
- The profile may be expressed, as an XML document whose contents may be incorporated into a contract (Paragraph 34).
- One example of a contract template is an almost complete electronic contract document with a few fields left blank (Paragraph 34).

However, Dan's approach leads to finalizing the structure of a contract document. Conversely, the Applicants' approach occurs after the structure of the XML is finalized between parties. Dan's approach entails filling a template with values. The template, even if it incorporates an XML, remains as a single structure. Conversely, in the Applicants' approach, the XML is broken down into fragments of several static and dynamic structures.

Next, the Applicants provide for classifying the dynamic structures of the XML transaction with empty tags and single occurrence classifiers for repeating dynamic structures. The Office Action suggests that Dan teaches this as follows:

- Negotiable field 1023 or 1024 may be treated as blank that may be completed by the negotiating party (Paragraph 35).

However, Dan shows the method of leaving few fields blank in a document template, while it is being exchanged between partners. In others words, the document is transmitted with blank values. Conversely, the Applicants' approach does not involve leaving fields blank in a transaction when exchanged between partners. Additionally, Dan talks only about blank fields.

Whereas, the Applicants provide a mechanism where an entire sub-structure within an XML can be left with empty tags with attributes embedded in the XML itself or as an attribute of the list indicating if the entire structure is a single or repeating occurrence. Furthermore, as indicated above, Dan's approach has no intelligence on whether the "blank" field is to be filled. Conversely, the Applicants mark the pieces of XML that are truly dynamic and provide a method for empty tags, and classifiers to indicate if the structure is single occurrence, multiple occurrences, etc.

Next, the Applicants provide for building a list of a sequence of static and dynamic structures. The Office Action suggests that Dan teaches this as follows:

- A set of sequencing rules 180 may be provided in meta contract 110 to ensure that the various negotiation actions are being issued in the correct order (Paragraph 32).

However, the Applicants list is different from that taught in Dan as described above. Moreover, Dan's use of sequencing rules is fundamentally different than the Applicants' approach. Dan's sequencing rules refer to the *choreography* of a series of negotiation actions as part of a long-running transaction. Dan himself defines the various Action Names in his Paragraph 42. This clearly demonstrates that Dan does not teach the Applicants' claimed invention. Moreover, Dan is using his sequencing rules to determine that the negotiation actions are being issued in the correct order. Whereas, the Applicants are simply building a "sequence of said static and dynamic structures." It appears the Office Action, in rejecting the Applicants' claims, simply has done a keyword search looking for the word "sequencing" and when found is using it to reject the Applicants' claims without reading the context of how Dan is using

“sequencing”. Page 24 of the Office Action states that one can broadly interpret the Applicants’ “building a sequence” as Dan’s “sequencing rules” since the sequence claimed in the independent claims is not defined in those claims. However, one can only broadly interpret claimed language to the extent that it is reasonable (see MPEP §904.01). In this case, one cannot simply isolate the words “building a sequence” and liken it to “sequencing rules.” Rather, one must read the entire phrase “building a list of a sequence of said static and dynamic structures” and compare that to “[a] set of sequencing rules 180 may be provided in meta contract 110 to ensure that the various negotiation actions are being issued in the correct order.” Clearly, Dan is not referring to its static and dynamic structures in this sentence of paragraph 32. Nor is Dan referring to building any type of list of the sequence. Rather, Dan is merely determining whether its negotiation actions are being issued in a correct order according to a set of sequencing rules. Accordingly, the type of argument being presented in the Office Action is improper.

Next, the Applicants provide for linking the list to a type of XML transaction and a predetermined trading partner profile. The Office Action suggests that Dan teaches this as follows:

- Starting definitions and values for these types of information in the negotiated contract may be provided in a TPA template or party profile (Paragraph 32).

Dan provides starting definitions and values from a TPA template or party profile. For example, Dan might derive starting values from the TPA template for Company ABC. However, the Applicants’ approach links the list of static and dynamic parts to the defined Trading Partner Profile and specific transaction type. For example, the Applicants could link the list to Company ABC and a shipping transaction type. Dan does not and cannot accomplish this.

Furthermore, the Applicants provide for combining the static structures with the dynamic structures at a runtime of the XML transaction based on the sequence, the type of XML transaction, the partner profile, and the dynamic structures of XML transaction. The Office Action suggests that Dan teaches this as follows:

- One example of a contract template is an almost complete electronic contract document with a few fields left blank (Paragraph 34).
- Once a contract template of Dan is sent for negotiation, it contains fields that are set and non-negotiable, and fields that are negotiable.

However, Dan teaches how to fill a form with some fields filled in and some fields left blank. Dan's teaching transmits a template with blank and non-blank fields. Moreover, Dan does not teach how to break an XML transaction, and classify them as static and dynamic components. Furthermore, Dan does not teach how to dynamically fill values in an XML to construct a complete XML prior to transmission. Rather, Dan teaches how to send blank tags to be optionally filled by receiving partners. Moreover, Dan's teaching does not assemble the complete XML (with no blank field/structures) prior to transmission. Dan's teaching involves two or more partners to fill the blank fields. Conversely, the Applicants' approach involves breaking and assembling the XML within a single partner's environment before transmitting the information to another partner. Moreover, the Applicants combine static and dynamic structures at runtime, with actual transaction values, at runtime, when the complete transaction is sent to the partner. Additionally, Dan's method talks about building an almost complete electronic contract document prior to runtime. In the Applicants' approach, actual transaction values can differ for each individual transaction (ex. shipped product, quantity and dollar value in today's transaction

can be different from the very next instance of the transaction for the same partner, and same transaction type).

Furthermore, the Applicants provide for pre-building of the static structures to occur prior to runtime of the XML transaction. The Office Action suggests that Dan teaches this as follows:

- The profile serves as the starting point of a negotiation by providing an initial version of a contract document (Paragraph 33).
- The profile may be expressed, as an XML document whose contents may be incorporated into a contract (Paragraph 34).
- One example of a contract template is an almost complete electronic contract document with a few fields left blank (Paragraph 34).
- Contract of Dan runs once the negotiation phase begins to fill in the initial blank negotiable fields 1023 and 1024.

However, as mentioned Dan teaches how to fill a form with some fields filled in and transmit a template with blank and non-blank fields. But, Dan does not teach how to break an XML transaction and classify the static components. Dan teaches how to fill fields in a template as part of the process of building and transmitting an instance of the XML. This entire process occurs once per document transmission. Conversely, the Applicants' approach involves two distinct events. In a first phase, the static component is classified and filled in. The phase first occurs one time for a partner and transaction type. The second phase occurs every time a transaction is sent to a partner, wherein the static structures are combined with the dynamic structures, according to the sequence defined by the list.

Next, the Applicants provide for filling the empty tags of the dynamic structures with business data values. The Office Action suggests that Dan teaches this as follows:

- Negotiable 1023 or 1024 may be treated as a blank that may be completed by the negotiating party (Paragraph 35).

However, again, Dan teaches how to fill a form with some fields filled in and transmit a template with blank and non-blank fields. Dan does not teach how to break an XML transaction and classify the static and dynamic components. Rather, Dan teaches how to fill fields in a template as part of the process of building and transmitting an instance of the XML. Again, this entire process occurs once per document transmission. Conversely, the Applicants' approach involves two distinct steps. In the first phase, the static component is classified and filled in. The first phase occurs one time for a partner and transaction type. The second phase occurs every time a transaction is sent to a partner, wherein the dynamic structures are automatically filled based on the associated pre-defined business logic, and the static structures are combined with the dynamic structures according to the sequence defined by the list. Moreover, the Applicants' approach provides the manner of filling actual business data values in the dynamic sections of an XML to construct a complete XML prior to transmission. Whereas, Dan teaches how to send blank tags to be optionally filled by receiving partners. Furthermore, the Applicants' approach provides for expanding dynamic structures (ex. multiple occurrences) at runtime, based on actual transaction business data values occurring at runtime.

Next, the Applicants provide for creating a copy of a pre-defined data type definition format comprising the XML format. The Office Action suggests that Thomas teaches this as follows:

- The processor reads 12 the document type definition (DTD) of the first XML file and creates copy 13 of the DTD.

However, the Applicants use the DTD to validate the final XML after combining static and dynamic structures, and filling actual transaction business data values at runtime. As previously mentioned, the manner of comparison and validation is different between Thomas the Applicants' claimed invention.

In view of the foregoing, the Applicant respectfully submits that the collective cited prior art do not teach or suggest the features defined by amended independent claims 1, 11, and 21 and as such, claims 1, 11, and 21 are patentable over Dan alone or in combination with Thomas. Further, dependent claims 2-6, 8-10, 12-16, 18-20, and 22-24 are similarly patentable over Dan alone or in combination with Thomas, not only by virtue of their dependency from patentable independent claims, respectively, but also by virtue of the additional features of the invention they define. Thus, the Applicant respectfully requests that these rejections be reconsidered and withdrawn. Moreover, the Applicant notes that all claims are properly supported in the specification and accompanying drawings. In view of the foregoing, the Examiner is respectfully requested to reconsider and withdraw the rejections.

II. Formal Matters and Conclusion

With respect to the rejections to the claims, the claims have been amended, above, to overcome these rejections. In view of the foregoing, the Examiner is respectfully requested to reconsider and withdraw the rejections to the claims.

In view of the foregoing, Applicants submit that claims 1-6, 8-16, and 18-24, all the claims presently pending in the application, are patentably distinct from the prior art of record and are in condition for allowance. The Examiner is respectfully requested to pass the above application to issue at the earliest possible time.

Should the Examiner find the application to be other than in condition for allowance, the Examiner is requested to contact the undersigned at the local telephone number listed below to discuss any other changes deemed necessary. Please charge any deficiencies and credit any overpayments to Attorney's Deposit Account Number 09-0456.

Respectfully submitted,

Dated: April 9, 2007

/Mohammad S. Rahman/
Mohammad S. Rahman
Registration No. 43,029

Gibb & Rahman, LLC
2568-A Riva Road, Suite 304
Annapolis, MD 21401
Voice: (301) 261-8625
Fax: (301) 261-8825
Customer Number: 29154